

EXHIBIT 15



Introduction to Persistent Uniform Resource Locators

Keith Shafer, Stuart Weibel, Erik Jul, Jon Fausey
 OCLC Online Computer Library Center, Inc.
 6565 Frantz Road, Dublin, Ohio 43017-3395

Exhibit
0010

Problem introduction

The point-and-click idiom of World Wide Web usage has made Internet browsing as easy as tapping on the door with your index finger, but every Net surfer soon learns that, too often, the summons remains unanswered. The now-familiar Uniform Resource Locator (URL) can change at the whim of hardware reconfiguration or file system reorganization or with changes in organizational structure, leaving users stranded in 404 limbo... *Document Not Found.*

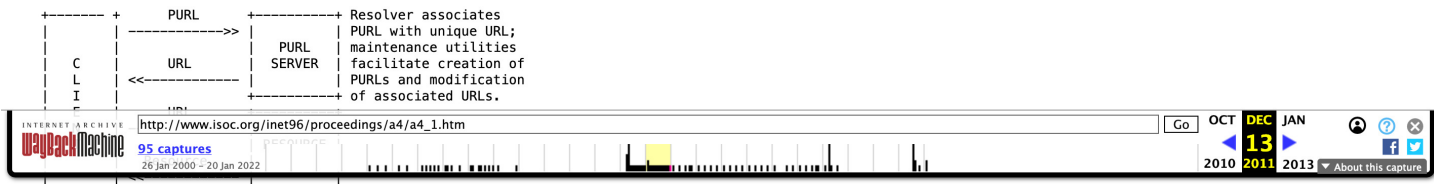
The unpredictable mobility of Internet resources is an inconvenience at best. For librarians, it is a serious problem that compromises their service to patrons and imposes an unacceptably large burden on cataloging. Putting all the pieces in place will require consensus in the IETF, developments in the community of web browser implementors, and deployment of new code by the community of network system managers who administer the Domain Name System (DNS) for the Internet. The concerns and needs of the library community may not be fully appreciated or adequately addressed by these groups in a timely manner. Libraries can and should provide leadership in the solution of these problems.

The persistence requirement of URN schemes is not a technological issue so much as an outcome of the social structures that evolve to meet a common community need. OCLC's origin is deeply rooted in precisely this shared commitment to providing reliable, long-term access to information.

Today's solution: Persistent URLs

To aid in the development and acceptance of URN technology, OCLC has deployed a naming and resolution service for general Internet resources. The names, which can be thought of as "Persistent URLs" (PURLs), can be used in documents, Web pages, and cataloging systems. PURLs increase the probability of correct resolution over that of URLs, and thereby reduce the burden and expense of maintaining viable, long-term access to electronic resources.

Functionally, a PURL is a URL. However, instead of pointing directly to the location of an Internet resource, a PURL points to an intermediate resolution service. The PURL Resolution Service associates the PURL with the actual URL and returns that URL to the client. The client can then complete the URL transaction in the normal fashion. In Web parlance, this is a standard HyperText Transfer Protocol (HTTP) *redirect*.



The redirection used in the OCLC PURL Service is a standard HTTP feature. Redirection keeps the OCLC PURL Server load light. For example, the single processor Sun4 file server currently running the OCLC PURL Service can support over 50 resolutions per second on a 500,000 PURL database. In addition to performing redirection, the PURL Server could serve all/some documents directly. The key to the PURL Server is *indirection*, not redirection--i.e., naming items to separate *location* from *identification*.

PURLs look just like URLs because they *are* URLs. A PURL has three parts: (1) a *protocol*, (2) a *resolver address*, and (3) a *name*. The following PURL examples use the same access protocol (http) to connect to the same PURL Resolver (purl.oclc.org) to resolve *different* names:

<http://purl.oclc.org/keith/home>
<http://purl.oclc.org/OCLC/PURL/FAQ>
<http://purl.oclc.org/OCLC/OLUC/32127398/1>

protocol resolver address name

Note that the *resolver address* is the IP address or domain name of the PURL Resolver. This portion of the PURL is resolved by the Domain Name Server (DNS). The *name* is user-assigned and is resolved by the associated PURL Resolver as described above.

PURLs provide the means of assigning a name for a network resource that is persistent, even if the item changes its actual location. For instance, when this paper was written, one of the author's home pages was located at <URL:<http://www.oclc.org:5046/~shafer>> and referred to by the PURL <URL:<http://purl.oclc.org/keith/home>>. If the home page later moves, only a single change to the PURL database is required and instances of the PURL in documents will remain valid. That is, if properly maintained, the PURL <URL:<http://purl.oclc.org/keith/home>> will always point to the current home page no matter where it is. Similarly, PURLs distributed in bibliographic records or by any other mechanism can remain viable over time without necessitating the maintenance task to all instances of the records.



Although one can change what a PURL resolves to, one cannot change the PURL itself. This means that a PURL can last longer than any particular URL that may be associated with it. PURLs persist indefinitely, and as long as they do, all instances of such PURLs (for example, links in a Web document or a bibliographic record) remain valid. Of course, someone has to operate the PURL Resolvers that provide this persistence. If the associated URL of a PURL becomes outdated, resolution of the PURL may fail. However, the PURL and its full history will be available as long as the PURL Service itself is maintained.

It is important to note that persistence is a function of organizations, not technology. It is expected that a PURL Service will always be available to resolve the PURL. They are called PURLs instead of URLs (even though they are URLs) to emphasize that PURLs have more support than normal URLs. For instance, they show a commitment by organizations running PURL Resolvers to make the names persistent. OCLC has long been committed to facilitating access to the world's information, and that commitment stands behind PURLs, too. It is expected that other organizations with similar commitments to provide long-term access to information will want to run PURL Servers as well: government agencies, publishers, libraries, and universities, for example.

PURLs and URNs

PURLs are a direct result of OCLC's work in the Uniform Resource Name (URN) standards and library cataloging communities. The assignment of PURLs is an intermediate step toward the time when URNs are an integral part of the Internet information architecture. The eventual syntax of URNs is clear enough at this time to afford confidence that the syntax of PURLs can be inexpensively and mechanically translated to the eventual URN form. For instance, the PURL

<http://purl.oclc.org/keith/home>

protocol resolver address name

could be written something like the following using the Path URN syntax:



where the URN is essentially a hierarchical name with the first portion representing the *naming authority* and the second portion signifying the *local name*. Note that the *naming authority* in this example is merely the *resolver address* of the corresponding PURL, written backwards (with slashes instead of dots) and the *local name* is the same.

Syntax aside, one way to think of URNs and PURLs is to consider what it would take to turn a PURL Server into a URN Server: very little. If Web browsers were changed to recognize the syntax *URN:/org/oclc/purl/keith/home* and could then connect to the resolver at *purl.oclc.org*, the resolver could then resolve the name */keith/home*.

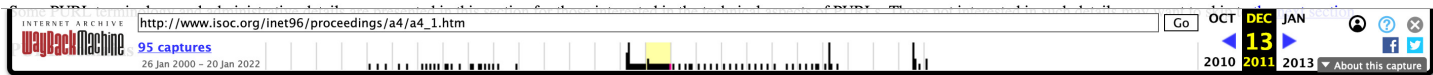
PURLs are not proprietary

PURLs are not a proprietary solution for naming. PURLs introduce no new protocols and require no client modifications. Instead, PURLs use standard HTTP protocols to connect to PURL Resolvers and standard HTTP redirects to return information to the requesting client.

It is expected that several PURL Resolvers will be running around the world with each one responsible for resolving its particular name space. Moreover, because PURLs are a distributed, not centralized, solution, the name portion of a PURL need not be globally unique. Clearly these PURLs use different resolvers:

<http://purl.oclc.org/keith/home>
<http://purl.fake.com/keith/home>

Some technical details



Domains are subdivisions of the name space on a PURL Resolver. They are very much like directories in a file system. The current implementation provides the ability to control who has write access (i.e., the ability to create PURLs, subdomains, etc.) within a domain. In the future, read access control will also be provided, but for now, anybody can read (i.e., resolve) anybody else's PURLs.

There are two varieties of domains:

- **Top-level domains**, as their name implies, occupy the top-level of the name space on a PURL Resolver. Users that own a top-level domain own and control access to that entire subdivision of the PURL Resolver's name space.
- **Subdomains** exist within top-level domains or other subdomains to any level of nesting. Users can create subdomains in any domain for which they have write privileges.

For example, the PURL <URL:<http://purl.fake.com/A/B/C/document>> has three domains, *A*, *B*, and *C*. *A* is a top-level domain, *B* is a subdomain of *A* and *C* is a subdomain of *B*.

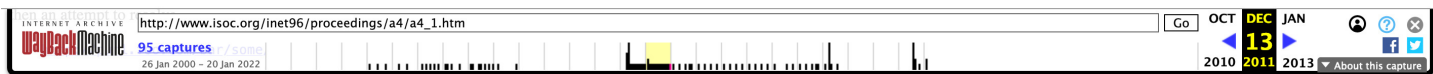
Partial redirection

The concept of partial redirection is the use of a domain as a prefix for a localized hierarchy of URLs. This is possible because a PURL Resolver will resolve as much of a PURL as it can find in its database and append the remainder (unresolved portion) to the end of the resolved URL. For example, if the PURL partial redirect

<http://purl.foo.com/bar/>

exists and is associated with the URL

<http://your.Web.server/your/Web/root/>



will resolve to the URL

<http://your.Web.server/your/Web/root/some/stuff.html>

Using this concept a partial redirect can serve as the permanent name prefix for all the resources stored at a Web site or any hierarchical subset thereof. Every document stored under the server's Web root directory can then be accessed by appending its relative (i.e., the *partial*) path to the partial redirect. This would allow the site's users to use the partial redirect as the prefix for all documents at the site. In addition, relocating the entire Web site would require changing only the single partial redirect; users of the site would see no changes.

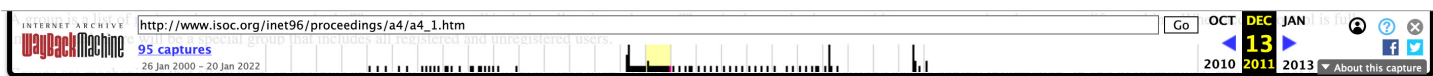
A partial redirect is a special-purpose PURL that acts like a domain. A regular domain has no associated URL. It is just part of a local name. Although a partial redirect has a URL associated with it, that URL is not guaranteed (or even expected) to reference an actual resource. The URL associated with a partial redirect may only be a prefix common to the complete URLs of multiple resources. In contrast, the URL associated with a PURL is expected to reference a single actual resource.

Registered users

Registered users of a PURL Resolver are people who have created a user ID and password on the PURL Resolver. To become a registered user, point a Web browser at a PURL Resolver and follow the resolver's instructions for becoming a registered user. The resolver should provide a form on which to enter a user ID and a password. Immediate confirmation will be given on the success or failure of the registration process.

Unregistered users can resolve and search a resolver for only *universally resolvable* PURLs, domains, and partial redirects. A privately resolvable PURL, domain, or partial redirect is one that will only be resolved for designated registered users of the PURL Resolver on which it resides. (At the time this document was written, all PURLs were universally resolvable because read access control was under development.)

Groups



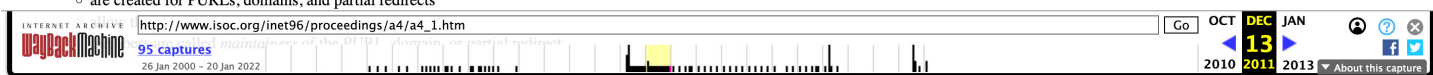
capabilities on their members. A group must be specified as a member of a PURL's, domain's, or partial redirect's read, write, or maintenance [access control list](#) to have any effect on its members.

Only registered users can create groups. To create a group, direct a Web browser to a PURL Resolver and follow the resolver's instructions for creating a group. The PURL Resolver should provide immediate feedback on the success of the group creation request.

Access control lists

An access control list is a list of [registered users](#), [groups](#), or both, that is associated with one and only one PURL, domain, or partial redirect. Access control lists are attributes of a PURL, domain, or partial redirect. There are three kinds of access control lists:

- **Read access control lists**
 - are created for PURLs, domains, and partial redirects
 - allow their members to resolve the PURL, domain, or partial redirect
 - members are called *readers* of the PURL, domain, or partial redirect
- **Write access control lists**
 - are created for domains
 - allow their members to create PURLs, domains, and partial redirects within domains
 - members are called *writers* of the domain
- **Maintenance control lists**
 - are created for PURLs, domains, and partial redirects



Creating PURLs

A registered user can create a PURL provided that:

- The [top-level domain](#) of the name exists. Top-level domains cannot be created automatically. They require a request for manual intervention performed by the PURL Resolver's administrator.
- The user has [write](#) access to the top-level domain used in the name.
- The user has write access to the last existing subdomain in the name or all the subdomains in the name do not exist (in which case they are created automatically).
- The name does not already exist.

The creator of a PURL assigns the name component of the PURL. Names can be arbitrary. There need be no relationship between the name in a PURL and the URL associated with it. A PURL can look very different from its associated URL. For example, the PURL

<http://purl.oclc.org/foo/bar>

can have the associated URL

http://my.address.org/very/long/path/name/and/obscure/file_name.txt

What resources should have PURLs?

Users should assign a PURL to any discrete resource for which reliable access over time is desired. For example, a home page, an electronic journal, an individual article, or a paper are good candidates for a persistent



also good candidates. Objects at the top of hierarchies of objects that might be moved as a unit are excellent partial redirect candidates. Depending on the underlying nature of the hierarchy, the lower-level objects may not require PURLs. For example, if the hypertext object hierarchy corresponds closely to the hierarchy of the objects in the underlying file system and the hypertext links are relative links based on the file system hierarchy, then a single PURL for the top-level object in the hierarchy is all that may be necessary.

To create a PURL, point a Web browser to a PURL Resolver and follow the resolver's instructions for creating a PURL. PURL Resolvers provide a form to fill out to create a PURL. This form should provide a default public domain with universal write access.

Maintaining PURLs

PURLs are not updated automatically when their associated URL changes unless some outside process is run to notify the corresponding PURL Resolver. A [maintainer](#) must update the information in the appropriate PURL Server when the associated URL changes. It is the responsibility of a PURL's owner and its maintainers to update the PURL when the associated URL changes.

PURL maintenance can be performed by connecting to a PURL Resolver using a Web browser and then using the PURL Resolver's maintenance forms to make the appropriate changes to the desired PURL. Only authorized PURL maintainers can modify a PURL.

A PURL maintainer can turn off PURL resolution. This is done by entering a new, *empty*, URL on the PURL maintenance form for the PURL in question. This will cause a *history page* to be returned when resolution of the PURL is attempted. A history page for a PURL contains administrative information accumulated over time and details about past associated URLs.

PURL maintenance is an important difference between PURLs and several URN proposals. In some URN proposals, a URN is a permanent name for a unique resource and only that resource forever. Some URN proposals would allow the same resource to move, but would not allow a different resource to be associated with the URN. PURLs make the name permanent, but allow the associated URL to change.



Users can select (click on) a PURL on a Web page or in a document, and the PURL should be resolved to the associated URL, which the browser will then use to access the resource.

Users can put PURLs in Web pages, documents, or other resources with confidence that the PURL will persist over time. The links will remain valid even if the associated URLs change. This does not mean that a PURL magically changes its own associated URL when the referenced resource moves--the maintainers of the PURL make this happen.

Users can submit a PURL (or even just part of a PURL) to the PURL Resolver to obtain more information about the PURL, the associated URL, or, in some cases, the resource itself.

The future

PURLs have been assigned to records cataloged in the Internet Cataloging Project, the U.S. Department of Education-funded project to advance cataloging practice for Internet resources. This project represents the leading edge of MARC description of Internet resources, and has become the forum for discussion and development of standard practice in this new area of digital librarianship. (See the InterCat home page at <URL:<http://purl.oclc.org/net/InterCat>>.)

The OCLC PURL Service has been running since the beginning of January 1996. As of March 28, 1996, it had serviced 178,000 resolution requests for the 5,500 PURLs in its local database for 5,000 different users.

Although a PURL Service is being run and maintained at OCLC, the PURL model lends itself to distribution across the Net. Since the introduction of the PURL model and services, a number of institutions have expressed an interest in running their own PURL Servers. OCLC freely distributes the PURL source code to aid in rapid, wide distribution of this enabling technology. See <URL: <http://www.oclc.org/oclc/purl/download>> for details.

PURLs are a direct result of ongoing [research](#) at OCLC Online Computer Library Center, Inc. [OCLC](#) is a nonprofit computer library service and research organization whose computer network and services link more than 21,000 libraries in 63 countries and territories.

For further PURL information, please see

- The OCLC PURL Service at <URL:<http://purl.oclc.org>>.
- The PURL demonstration page at <URL:<http://purl.oclc.org/OCLC/PURL/demo>>.
- The PURL FAQ at <URL:<http://purl.oclc.org/OCLC/PURL/FAQ>>, and
- The PURL-L mailing list described at <URL:<http://purl.oclc.org/OCLC/PURL/PURL-L>>.

Additional PURL project staff members include [Eric Miller](#), [Roger Thompson](#), [Vince Tkac](#), and [many others](#).